

Tematy: [Zawartość generowana](#) | [Liczniki](#) | [Cytaty](#)

Druga specyfikacja CSS pozwala na generowanie zawartości witryny oraz dodawanie liczników dla paragrafów lub całych bloków tekstu. Niestety rozwiązania te nadal nie są obsługiwane przez przeglądarkę MS Internet Explorer.

Zawartość generowana

Do stworzenia zawartości generowanej niezbędne jest zastosowanie następujących elementów:

-

pseudoelementów :after i :before,

-

reguły content.

Na rysunku 14.1 zamieściłem przykład działania pseudoelementów :after i :before dla posiadaczy MS Internet Explorera oraz innych przeglądarek, które nie radzą sobie z tym zagadnieniem.

Rysunek 14.1. Przykład działania zawartości generowanej

Pseudoelementy :after i :before

Zawartość generowana jest realizowana w oparciu o opisane wcześniej pseudoelementy :before i :after oraz polecenie content. Więcej informacji na temat możliwości oferowanych przez polecenie content znajduje się w dalszej części niniejszego rozdziału.

Poniżej znajduje się prosty przykład, w którym:

-

do nagłówków H1 dodano informację przed główną treścią;

-

do nagłówków H2 dodano informację po głównej treści.

H1:before

{

content:"tutaj pojawi się licznik. ";

color:red;

font-size:12pt;

}

H2:after

{

Zawartość generowana i liczniki

Dodał Administrator
czwartek, 11 marzec 2010 15:31

```
content:&quot;; - to koniec nagł&ouml;wka&quot;;  
  
color:red;  
  
font-size:12pt;  
  
}  
  
<<przyk&ouml;lad>>  
  
<html>  
  
<head>  
  
<meta http-equiv=&quot;Content-type&quot; content=&quot;text/html;charset=ISO-8859-2&quot;>  
  
<title>Tytuł Strony WWW</title>  
  
<style type=&quot;text/css&quot;>  
  
H1:before  
  
{  
  
content:&quot;tutaj pojawi się licznik. &quot;;  
  
color:red;  
  
font-size:12pt;  
  
}  
  
H2:after  
  
{  
  
content:&quot;; - to koniec nagł&ouml;wka&quot;;  
  
color:red;  
  
font-size:12pt;  
  
}  
  
</style>  
  
</head>  
  
<body>  
  
<h1>Nagł&ouml;wek H1</h1>  
  
<h2>Nagł&ouml;wek H2</h2>  
  
</body>
```

</html>

<</przykład>> **Reguła content**

content

Dostępne wartości:

normal | none | "dowolna treść" | counter | open-quote | close-quote | no-open-quote | no-close-quote | inherit

Wartość domyślna:

Normal

Przykład:

```
h1:after {content:"informacja widoczna przed nagłówkiem";}
```

Dziedziczenie:

Nie

Obsługa:

Opera, Mozilla Firefox, Chrome, Safari

Reguła content w połączeniu z pseudoelementami :after i :before daje możliwość określenia, co pojawi się przed lub po formatowanym elemencie. Najprostsze rozwiązanie umożliwia dołączenie dowolnej treści do formatowanych elementów. Wystarczy żądany tekst umieścić w cudzysłowie i na tym koniec.

H1:before

```
{  
  content:&quot;tutaj pojawi się licznik. &quot;;  
}
```

<</przykład>>

<html>

<head>

```
<meta http-equiv=&quot;Content-type&quot; content=&quot;text/html;charset=ISO-8859-2&quot;>
```

```
<title>Tytuł Strony WWW</title>
```

Dodał Administrator
czwartek, 11 marzec 2010 15:31

```
<style type="text/css">
H1:before
{
content:"tutaj pojawi się licznik. ";
color:red;
font-size:12pt;
}
H2:after
{
content:" - to koniec nagłówka";
color:red;
font-size:12pt;
}
</style>
</head>
<body>
<h1>Nagłówek H1</h1>
<h2>Nagłówek H2</h2>
</body>
</html>
<</przykład>>
```

Oczywiście content kryje w sobie znacznie więcej możliwości, które zostały szerzej opisane w kolejnych podrozdziałach. **Liczniki**

W drugiej specyfikacji CSS wprowadzono pojęcie licznika oraz towarzyszące mu style. Dzięki temu rozwiązaniu można obecnie automatycznie ponumerować nagłówki lub bloki tekstu w dokumencie bez potrzeby tworzenia z nich list. Przygotowanie licznika składa się z trzech etapów, które zostały opisane w kolejnych podrozdziałach.

Na rysunku 14.2 zamieściłem przykład działania licznika. Obrazek jest przeznaczony dla posiadaczy MS Internet Explorera oraz innych przeglądarek, które nie radzą sobie z tym zagadnieniem.

Rysunek 14.2. Przykład działania licznika

Resetowanie licznika

counter-reset

Dostępne wartości:

none | nazwa licznika

Wartość domyślna:

none

Przykład:

```
BODY {counter-reset:licznik;}
```

Dziedziczenie:

Nie

Obsługa:

Opera, Mozilla Firefox, Chrome, Safari

Tworzenie licznika zaczyna się od wstawienia deklaracji odpowiedzialnej za jego wyzerowanie. Deklaracja taka powinna zostać przypisana do elementów nadrzędnych dla tego, w którym licznik ma działać. Poniżej znajduje się prosty przykład.

BODY

```
{  
  
counter-reset:licznik;  
  
}
```

Należy zwrócić uwagę na to, że bezpośrednio po poleceniu counter-reset występuje nazwa licznika, do której odnosi się rozkaz wyzerowania licznika. Nazwa jest dowolnie tworzona przez autora witryny. Dzięki takiemu rozwiązaniu można na jednej stronie uruchomić kilka lub kilkanaście niekolidujących ze sobą liczników.

Określanie wielkości wzrostu wartości

counter-increment

Dostępne wartości:

Zawartość generowana i liczniki

Dodał Administrator
czwartek, 11 marzec 2010 15:31

none | nazwa licznika + wartość wzrostu

Wartość domyślna:

none

Przykład:

```
h1:before {counter-increment:licznik;}
```

Dziedziczenie:

Nie

Obsługa:

Opera, Mozilla Firefox, Chrome, Safari

Drugim ważnym elementem jest zdefiniowanie wartości, o którą będzie wzrastać wskazanie licznika. Do tego celu służy polecenie counter-increment. Poniżej znajduje się prosty przykład zwiększający wartość licznika o 1.

```
H1:before
```

```
{  
  counter-increment:licznik;  
}
```

Natomiast kolejny przykład zwiększa wartość licznika o 3.

```
H1:before
```

```
{  
  counter-increment:licznik 3;  
}
```

<<przykład>>

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-type" content="text/html;charset=ISO-8859-2">
```

```
<title>Tytuł Strony WWW</title>
```

```
<style type="text/css">
```

Dodał Administrator
czwartek, 11 marzec 2010 15:31

BODY

```
{  
counter-reset:licznik;  
counter-reset:licznikdwa;  
}
```

H1:before

```
{  
content:counter(licznik) &quot;. &quot;;  
counter-increment:licznik;  
font-weight:bold;  
}
```

H2:before

```
{  
content:counter(licznikdwa) &quot;. &quot;;  
counter-increment:licznikdwa 3;  
font-weight:bold;  
}
```

</style>

</head>

<body>

<h1>Pierwszy nagłówek H1</h1>

<h1>Drugi nagłówek H1</h1>

<h1>Trzeci nagłówek H1</h1>

<h1>Czwarty nagłówek H1</h1>

<h1>Piąty nagłówek H1</h1>

<hr>

<h2>Pierwszy nagłówek H2</h2>

<h2>Drugi nagłówek H2</h2>

Dodał Administrator
czwartek, 11 marzec 2010 15:31

```
<h2>Trzeci nagłówek H2</h2>
```

```
<h2>Czwarty nagłówek H2</h2>
```

```
<h2>Piąty nagłówek H2</h2>
```

```
</body>
```

```
</html>
```

```
<</przykład>>
```

Należy zwrócić uwagę na to, że bezpośrednio po poleceniu counter-increment występuje nazwa licznika, do której odnosi się rozkaz wyzerowania licznika. Nazwa jest dowolnie tworzona przez autora witryny. Dzięki takiemu rozwiązaniu można na jednej stronie uruchomić kilka lub kilkanaście niekolidujących ze sobą liczników.

Podłączenie licznika

Ostatnim krokiem mającym na celu uruchomienie automatycznej numeracji elementów na stronie jest podłączenie licznika pod wybrane fragmenty strony. W tym wypadku przyda się reguła content, o której wspominałem wcześniej.

Jedną z wartości występujących wraz z content jest licznik. Odpowiednia deklaracja stylu włączającego licznik dla nagłówków h1 ma następującą postać:

```
H1:before
```

```
{  
  
content:counter(licznik) " " ;  
  
counter-increment:licznik ;  
  
font-weight:bold ;  
  
color:red ;  
  
}
```

Należy zwrócić uwagę na to, że bezpośrednio po wartości counter w nawiasie znajduje się nazwa licznika. Nazwa jest tworzona przez autora witryny i może przybierać dowolną postać. Ważne jest to, aby używać tej samej nazwy dla jednego licznika. Dzięki takiemu rozwiązaniu można na jednej stronie uruchomić kilka lub kilkanaście niekolidujących ze sobą liczników. Bezpośrednio po nawiasie można w cudzysłowie umieścić znak rozdzielający numer od numerowanego elementu, np. kropkę.

Wygląd licznika może być dowolnie formatowany. Wystarczy odpowiednie deklaracje stylów umieścić w sekcji aktywującej licznik. W przypadku mojego przykładu będzie to h1:before.

Poniżej zamieściłem przykład działającego licznika.

```
<<przykład>>
```

```
<html>
```

```
<head>
```


Zawartość generowana i liczniki

Dodał Administrator
czwartek, 11 marzec 2010 15:31

```
<meta http-equiv="Content-type" content="text/html;charset=ISO-8859-2">
```

```
<title>Tytuł Strony WWW</title>
```

```
<style type="text/css">
```

```
BODY
```

```
{  
  counter-reset:licznik;  
  counter-reset:licznikdwa;  
}
```

```
H1:before
```

```
{  
  content:counter(licznik, upper-roman) " . ";  
  counter-increment:licznik;  
  font-weight:bold;  
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1>Pierwszy nagłówek H1</h1>
```

```
<h1>Drugi nagłówek H1</h1>
```

```
<h1>Trzeci nagłówek H1</h1>
```

```
<h1>Czwarty nagłówek H1</h1>
```

```
<h1>Piąty nagłówek H1</h1>
```

```
</body>
```

```
</html>
```

```
<</przykład>>
```

Ciekawostką jest, że wykorzystując wartości stylu list-style-type, można wpływać na to, czy licznik wykorzystuje numerację rzymską, czy np. arabską. Poniżej przykład zmodyfikowanego arkusza stylów.

```
H1:before
```

Zawartość generowana i liczniki

Dodał Administrator
czwartek, 11 marzec 2010 15:31

```
{
content:counter(licznik, upper-roman) &quot;. &quot;;
counter-increment:licznik;
font-weight:bold;
}
<<przykład>>
<html>
<head>
<meta http-equiv=&quot;Content-type&quot; content=&quot;text/html;charset=ISO-8859-2&quot;>
<title>Tytuł Strony WWW</title>
<style type=&quot;text/css&quot;>
BODY
{
counter-reset:licznik;
}
H1:before
{
content:counter(licznik, upper-roman) &quot;. &quot;;
counter-increment:licznik;
font-weight:bold;
}
</style>
</head>
<body>
<h1>Pierwszy nagłówek H1</h1>
<h1>Drugi nagłówek H1</h1>
<h1>Trzeci nagłówek H1</h1>
<h1>Czwarty nagłówek H1</h1>
```

Dodał Administrator
czwartek, 11 marzec 2010 15:31

```
<h1>Piąty nagłówek H1</h1>
```

```
</body>
```

```
</html>
```

```
<</przykład>> Cytaty
```

quotes

Dostępne wartości:

none | "znak" "znak";

Wartość domyślna:

wg ustawień przeglądarki

Przykład:

```
q:before {quotes:"<<" ">>"};
```

Dziedziczenie:

Tak

Obsługa:

Mozilla Firefox, Opera

Za pomocą stylu `quote` można określić, w jaki sposób zostanie wyróżniony cytat zamieszczony w dokumencie. Bezpośrednio po deklaracji stylu w cudzysłowie należy zamieścić otwierający znak wyróżniający cytowany tekst. Następnie w ten sam sposób należy określić znak zamykający. Poniżej znajduje się przykładowy arkusz stylów.

Q

```
{
```

```
quotes:&quot;<<&quot; &quot;>>&quot;;
```

```
}
```

Korzystając z pseudolementów `:before` oraz `:after`, możemy kontrolować obecność znaków na początku i końcu cytowanego tekstu. Wystarczy za pomocą reguły `content` włączyć lub wyłączyć znaki cytatu. Poniżej przykład, w którym włączyłem znaki na początku i na końcu cytatu.

```
<<przykład>>
```

Dodał Administrator
czwartek, 11 marzec 2010 15:31

<html>

<head>

<meta http-equiv="Content-type" content="text/html;charset=ISO-8859-2">

<title>Tytuł Strony WWW</title>

<style type="text/css">

Q

{

quotes: "<" ">";

}

Q:before

{

content:open-quote;

}

Q:after

{

content:close-quote;

}

</style>

</head>

<body>

<p>

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec vehicula rhoncus nisl. Vivamus nec augue. Nulla a pede. Cras non quam a elit pretium eleifend. Integer ut quam. Proin dictum libero id ipsum. Integer dolor mi, varius eget, aliquet vel, ullamcorper ac, lectus. Phasellus ut risus ac metus posuere viverra. Cras pede justo, rutrum egestas, ultricies quis, gravida a, mauris. Aenean mi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. In vestibulum nunc in massa. Cras congue lorem sed libero. Sed sagittis volutpat arcu.

</p>

<q>

Phasellus quis enim sed velit pellentesque eleifend. Phasellus diam lorem, feugiat non, rhoncus et, pharetra id, lorem. Vestibulum fermentum odio sodales libero. Etiam a odio. Donec vitae mi venenatis augue consequat

Dodał Administrator
czwartek, 11 marzec 2010 15:31

imperdiet. Pellentesque molestie convallis lectus. Phasellus consetetuer fringilla magna. Vivamus ac est sed tellus lacinia dignissim. Aliquam elit. Duis orci. Cras vel velit sit amet erat consequat facilisis. Suspendisse sit amet lorem. Integer nonummy, purus sed laoreet commodo, dui risus pulvinar purus, vel blandit ligula lorem vel nibh. Nam risus diam, cursus vitae, accumsan et, porttitor id, turpis. Praesent a nulla. Mauris eget diam et felis facilisis tristique. Sed commodo tellus. In non urna. In porta dapibus neque. Sed cursus dapibus urna.

```
</q>
```

```
</body>
```

```
</html>
```

```
<</przykład>>
```

Kolejny przykład pokazuje cytaty, w którym wyłączyłem znaki rozpoczynające i kończące cytowany tekst.

```
Q:before
```

```
{  
  content:no-open-quote;  
}
```

```
Q:after
```

```
{  
  content:no-close-quote;  
}
```

```
<<przykład>>
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-type" content="text/html;charset=ISO-8859-2">
```

```
<title>Tytuł Strony WWW</title>
```

```
<style type="text/css">
```

```
Q
```

```
{  
  quotes: ";<&quot; &quot;>&quot;;  
  color:green;  
  font-style:italic;  
}
```

Dodał Administrator
czwartek, 11 marzec 2010 15:31

Q:before

```
{  
  
content:no-open-quote;  
  
}
```

Q:after

```
{  
  
content:no-close-quote;  
  
}
```

</style>

</head>

<body>

<p>

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec vehicula rhoncus nisl. Vivamus nec augue. Nulla a pede. Cras non quam a elit pretium eleifend. Integer ut quam. Proin dictum libero id ipsum. Integer dolor mi, varius eget, aliquet vel, ullamcorper ac, lectus. Phasellus ut risus ac metus posuere viverra. Cras pede justo, rutrum egestas, ultricies quis, gravida a, mauris. Aenean mi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. In vestibulum nunc in massa. Cras congue lorem sed libero. Sed sagittis volutpat arcu.

</p>

<q>

Phasellus quis enim sed velit pellentesque eleifend. Phasellus diam lorem, feugiat non, rhoncus et, pharetra id, lorem. Vestibulum fermentum odio sodales libero. Etiam a odio. Donec vitae mi venenatis augue consequat imperdiet. Pellentesque molestie convallis lectus. Phasellus consectetur fringilla magna. Vivamus ac est sed tellus lacinia dignissim. Aliquam elit. Duis orci. Cras vel velit sit amet erat consequat facilisis. Suspendisse sit amet lorem. Integer nonummy, purus sed laoreet commodo, dui risus pulvinar purus, vel blandit ligula lorem vel nibh. Nam risus diam, cursus vitae, accumsan et, porttitor id, turpis. Praesent a nulla. Mauris eget diam et felis facilisis tristique. Sed commodo tellus. In non urna. In porta dapibus neque. Sed cursus dapibus urna.

</q>

</body>

</html>

<</przykład>>